

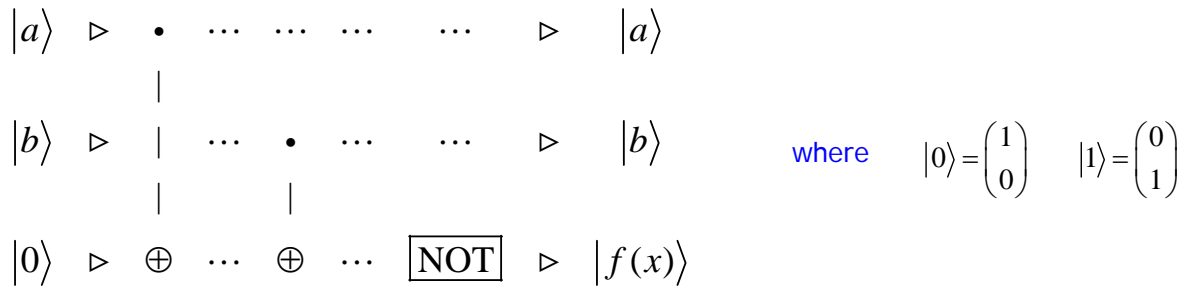
# A Quantum Circuit for Parallel Computation

Frank Rioux  
Emeritus Professor of Chemistry  
CSB|SJU

This tutorial deals with quantum function evaluation and parallel computation. The example is taken from pages 94-95 of *Exploring the Quantum* by Haroche and Raimond. A certain function of  $x$  yields the following table of results.

$$\begin{pmatrix} x & 0 & 1 & 2 & 3 \\ f(x) & 1 & 0 & 0 & 1 \end{pmatrix}$$

In the quantum circuit provided by Haroche and Raimond the top two wires carry the values of  $x$  in binary code (2 bits, 4 values) and the third wire is initially set to  $|0\rangle$ . After operation of the circuit on the initial state, the value of the function appears on the third wire. The qubits on the top wires are unchanged by the circuit. The circuit consists of a CnNOT gate employing all three wires, followed by a CNOT gate involving the second and third wires, and finally a NOT gate operating on the third wire.



First we establish that this circuit yields the results shown in the table, and then demonstrate that it also carries out a parallel calculation in one step using all four input values of  $x$ . The truth tables for the various quantum gates are provided in the Appendix.

$ 000\rangle$	$\xrightarrow{CnNOT}$	$ 0\rangle 00\rangle$	$\xrightarrow{I\otimes CNOT}$	$ 00\rangle 0\rangle$	$\xrightarrow{I\otimes I\otimes NOT}$	$ 00\rangle 1\rangle$	$f(0) = 1$
$ 010\rangle$		$ 0\rangle 10\rangle$		$ 01\rangle 1\rangle$		$ 01\rangle 0\rangle$	$f(1) = 0$
$ 100\rangle$		$ 1\rangle 01\rangle$		$ 10\rangle 1\rangle$		$ 10\rangle 0\rangle$	$f(2) = 0$
$ 110\rangle$		$ 1\rangle 11\rangle$		$ 11\rangle 0\rangle$		$ 11\rangle 1\rangle$	$f(3) = 1$

A superposition of all four input states behaves as follows under the operation of the quantum circuit.

$$\frac{1}{2} \begin{pmatrix} |00\rangle|0\rangle \\ + \\ |01\rangle|0\rangle \\ + \\ |10\rangle|0\rangle \\ + \\ |11\rangle|0\rangle \end{pmatrix} \xrightarrow{CnNOT} \frac{1}{2} \begin{pmatrix} |0\rangle|00\rangle \\ + \\ |0\rangle|10\rangle \\ + \\ |1\rangle|01\rangle \\ + \\ |1\rangle|11\rangle \end{pmatrix} \xrightarrow{I\otimes CNOT} \frac{1}{2} \begin{pmatrix} |00\rangle|0\rangle \\ + \\ |01\rangle|1\rangle \\ + \\ |10\rangle|1\rangle \\ + \\ |11\rangle|0\rangle \end{pmatrix} \xrightarrow{I\otimes I\otimes NOT} \frac{1}{2} \begin{pmatrix} |00\rangle|1\rangle \\ + \\ |01\rangle|0\rangle \\ + \\ |10\rangle|0\rangle \\ + \\ |11\rangle|1\rangle \end{pmatrix} \xrightarrow{Decimal} \frac{1}{2} \begin{pmatrix} |0\rangle|1\rangle \\ + \\ |1\rangle|0\rangle \\ + \\ |2\rangle|0\rangle \\ + \\ |3\rangle|1\rangle \end{pmatrix}$$

Haroche and Raimond describe this process as follows: "By superposing the inputs of a computation, one operates the machine 'in parallel', making it compute simultaneously all the values of a function and keeping its state, before any final bit detection is performed, suspended in a coherent superposition of all the possible outcomes."

It appears that we get four calculations in a single operation of the circuit. However, there's a catch, and that is that any inquiry into the results of the calculation can yield only one of the results. On measurement the quantum superposition collapses, as it always does, to one value and the others are irretrievably lost. As Haroche and Raimond write, "It is, however, one thing to compute potentially at once all the values of  $f$  and quite another to be able to exploit this quantum parallelism and extract from it more information than from a mundane classical computation. The final stage of information acquisition must always be a measurement."

In summary, this tutorial has demonstrated how a quantum circuit can function as an algorithm for the evaluation of a mathematical function, and how the same algorithm is capable of parallel evaluations of that function. The exploitation of quantum parallelism for practical outcomes such as factorization requires more elaborate quantum circuits than the one presented here.

## Appendix

NOT	CNOT	CnNOT																																																																						
$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	<table style="border: none; margin: auto;"> <thead> <tr> <th style="padding: 5px;">Decimal</th> <th style="padding: 5px;">Binary</th> <th style="padding: 5px;">'</th> <th style="padding: 5px;">Binary</th> <th style="padding: 5px;">Decimal</th> </tr> </thead> <tbody> <tr><td style="padding: 5px;">0</td><td style="padding: 5px;">00</td><td style="padding: 5px;">'</td><td style="padding: 5px;">00</td><td style="padding: 5px;">0</td></tr> <tr><td style="padding: 5px;">1</td><td style="padding: 5px;">01</td><td style="padding: 5px;">'</td><td style="padding: 5px;">01</td><td style="padding: 5px;">1</td></tr> <tr><td style="padding: 5px;">2</td><td style="padding: 5px;">10</td><td style="padding: 5px;">'</td><td style="padding: 5px;">11</td><td style="padding: 5px;">3</td></tr> <tr><td style="padding: 5px;">3</td><td style="padding: 5px;">11</td><td style="padding: 5px;">'</td><td style="padding: 5px;">10</td><td style="padding: 5px;">2</td></tr> </tbody> </table>	Decimal	Binary	'	Binary	Decimal	0	00	'	00	0	1	01	'	01	1	2	10	'	11	3	3	11	'	10	2	<table style="border: none; margin: auto;"> <thead> <tr> <th style="padding: 5px;">Decimal</th> <th style="padding: 5px;">Binary</th> <th style="padding: 5px;">'</th> <th style="padding: 5px;">Binary</th> <th style="padding: 5px;">Decimal</th> </tr> </thead> <tbody> <tr><td style="padding: 5px;">0</td><td style="padding: 5px;">000</td><td style="padding: 5px;">'</td><td style="padding: 5px;">000</td><td style="padding: 5px;">0</td></tr> <tr><td style="padding: 5px;">1</td><td style="padding: 5px;">001</td><td style="padding: 5px;">'</td><td style="padding: 5px;">001</td><td style="padding: 5px;">1</td></tr> <tr><td style="padding: 5px;">2</td><td style="padding: 5px;">010</td><td style="padding: 5px;">'</td><td style="padding: 5px;">010</td><td style="padding: 5px;">2</td></tr> <tr><td style="padding: 5px;">3</td><td style="padding: 5px;">011</td><td style="padding: 5px;">'</td><td style="padding: 5px;">011</td><td style="padding: 5px;">3</td></tr> <tr><td style="padding: 5px;">4</td><td style="padding: 5px;">100</td><td style="padding: 5px;">'</td><td style="padding: 5px;">101</td><td style="padding: 5px;">5</td></tr> <tr><td style="padding: 5px;">5</td><td style="padding: 5px;">101</td><td style="padding: 5px;">'</td><td style="padding: 5px;">100</td><td style="padding: 5px;">4</td></tr> <tr><td style="padding: 5px;">6</td><td style="padding: 5px;">110</td><td style="padding: 5px;">'</td><td style="padding: 5px;">111</td><td style="padding: 5px;">7</td></tr> <tr><td style="padding: 5px;">7</td><td style="padding: 5px;">111</td><td style="padding: 5px;">'</td><td style="padding: 5px;">110</td><td style="padding: 5px;">6</td></tr> </tbody> </table>	Decimal	Binary	'	Binary	Decimal	0	000	'	000	0	1	001	'	001	1	2	010	'	010	2	3	011	'	011	3	4	100	'	101	5	5	101	'	100	4	6	110	'	111	7	7	111	'	110	6
Decimal	Binary	'	Binary	Decimal																																																																				
0	00	'	00	0																																																																				
1	01	'	01	1																																																																				
2	10	'	11	3																																																																				
3	11	'	10	2																																																																				
Decimal	Binary	'	Binary	Decimal																																																																				
0	000	'	000	0																																																																				
1	001	'	001	1																																																																				
2	010	'	010	2																																																																				
3	011	'	011	3																																																																				
4	100	'	101	5																																																																				
5	101	'	100	4																																																																				
6	110	'	111	7																																																																				
7	111	'	110	6																																																																				